END TO END ADAPTATION FOR THE WEB
Matching Content to Client Connections

Kristoffer Getchell, Martin Bateman, Colin Allison, Alan Miller
School of Computer Science, The University of St Andrews, St Andrews, FIFE, SCOTLAND

Keywords: Internet, monitoring, quality of service, adaptation, media, perception.

Abstract: The size and heterogeneity of the Internet means that the bandwidth available for a particular download may range from many megabits per second to a few kilobits. Yet Web Servers today provide a one size fits all service and consequently the delay experienced by users accessing the same Web Page may range from a few milliseconds to minutes. This paper presents a framework for making Web Servers aware of the Quality of Service that is likely to be available for a user session, by utilising measurements of past traffic conditions. The Web Server adapts the fidelity of content delivered to users in order to control the delay experienced and thereby optimise the browsing experience. Where high bandwidth connectivity and low congestion exist high fidelity content will be delivered, where the connectivity is low bandwidth, or the path congested, lower fidelity content will be served, and delay controlled.

1 INTRODUCTION

Access bandwidths now range from a few kilobits per second through a mobile phone up to gigabits per second through Local Area Networks. Despite the bandwidth at the core of the Internet increasing, many bottlenecks continue to exist within the system; even users with high bandwidth connectivity may experience significant delay when engaging in network communication. At present the Internet relies upon the congestion control mechanisms embedded in TCP (Jacobson, 1988) to prevent congestion collapse (Jain and Ramakrishnan, 1988), which would render it unusable, yet applications are in a strong position to know how best to react to the onset of congestion (Floyd et al., 1997). If an application was made congestion aware it could directly control the amount of data sent. If congestion was high it could reduce the absolute number of bytes transmitted as well as the rate at which they were sent.

This paper addresses two questions: How is it possible for an application to become aware of network conditions and, given this awareness, how can a system be designed to allow application led adaptation to occur. A framework, which consists of three components, is proposed. A Network Monitor provides a Server with measurements of the Quality of Service (QoS) that the network is providing. A Network Aware Web Server handles the dynamic decisions required in order to determine the optimal version of a site to send to a connecting client and a Content Adaptation Tool allows content providers to generate, from a single high quality site, versions that are appropriate for different connection types.

From the user perspective a number of factors contribute to the perceived QoS offered, with both the speed of presentation and the quality of resources being important. A fast loading, simple Web Site is not necessarily going to be considered to be high quality by all users, and neither is a slow loading, highly detailed one (Bouch and Sasse, 1999, Bouch et al., 2000, Bouch and Sasse, 2000). From the content provider's perspective, a trade-off must be met whereby a balance between the speed at which resources can be provided to the target audience and the quality of the resources provided is achieved. Currently this trade-off is managed offline with content providers producing Web Sites which will be viewable within reasonable periods of time by the majority of Internet users. Those Internet users with connections slower than the speeds accounted for during the development of a site will be left with a poor browsing experience, whilst those with faster connections could have been supplied with pages of a higher fidelity. As the diversity of connection technologies continues to expand, this disparity is set to grow yet further.

By considering the QoS offered to the browsing client by the network, an adaptation framework

Getchell K., Bateman M., Allison C. and Miller A. (2005).
END TO END ADAPTATION FOR THE WEB - Matching Content to Client Connections.
In *Proceedings of the Second International Conference on e-Business and Telecommunication Networks*, pages 131-136
DOI: 10.5220/0001416901310136

would afford content providers the ability to focus on producing the highest quality resources, without having to consider the limitations that slow network links may cause. Lower fidelity resources can then be generated automatically using adaptation tools.

This paper continues with a discussion of related work in section 2 which leads to a discussion of the QoS issues of relevance to this work. Section 4 describes the framework used to allow the Network Aware Server to modify its behaviour as described in section 5. An evaluation of the framework thus far is provided in section 6, with section 7 concluding and drawing the paper to a close.

2 RELATED WORK

Numerous studies have advocated the adaptation of content delivered in response to the load placed upon a Web Server. A similar approach to those outlined in (Pradhan and Claypool, 2002, Barnett, 2002) is adopted by our work. However we aim to address the case where the total load on the Server is manageable, but specific users are receiving poor service because of congestion or connectivity issues.

With the expansion of e-commerce, the ways in which people interact with online vendors is of great importance to business sectors. (Bhatti et al., 2000) discuss the issues surrounding user tolerance, with regards to the levels of service offered, reiterating the importance of user perception. Through experimentation the level of user satisfaction is tested, with the results showing that if a user is provided with some feedback relatively quickly, they are generally satisfied to wait for extended periods while the remainder of their request is fulfilled.

Taking the concept of adaptation to mobile devices, (Abdelzaber and Bhatti, 1999, Shaha et al., 2001) discuss the issues surrounding the need to augment our traditional understanding of QoS in order to make the best use of a new wave of access device. Existing services designed for PC delivery are not well suited to PDAs. Owing to the poor quality of PDA delivered content it is hardly surprising that the mobile device users are often left frustrated and alienated. Media adaptation techniques however, may help; allowing mobile users to more readily gain access to the QoS adapted versions of a particular resource.

3 QUALITY OF SERVICE ISSUES

The aim of this work is to provide mechanisms that can maximise the QoS perceived by users for a given set of network conditions. In order to do so it is necessary to establish metrics for both the user perceived and network observed QoS.

3.1 User Perceived QoS

There is no single mapping between network level metrics and the way in which a user perceives QoS. Here we identify three factors that contribute to whether a user perceives a Web Browsing session as satisfying or not.

1. **Fidelity**: The quality of data sent to clients is vitally important – if the technical quality is too low (i.e. videos are blurry, sound is skewed, images are fuzzy or too small etc) then the user will perceive poor resource quality. However if the technical quality is too high (i.e. videos and sounds are jumpy because there are delays in getting data to the client, images are too large or too slow to load etc) then the user will again perceive poor quality.
2. **Delay and Feedback**: When a user is interacting with a Web Site, they require feedback for their actions. If it takes 2 minutes to load each page on a site, then the length of time required in order to supply feedback to the user (i.e. the next page they asked for) would be excessive and so user-perceived quality would suffer. However if the pages load too quickly, then it is possible that the user will not recognise the true speeds – delays of anything under 30ms are almost unnoticed by users and as such resources should not be wasted trying to reduce delays below 50ms or so (Abdelzaber and Bhatti, 1999).
3. **Consistency:** If there is consistency of presentation within a session the user will perceive a higher QoS during their browsing session (Bouch and Sasse, 2000).

3.2 Network QoS

QoS is usually quantified at the network level. At this level there are two challenges; how to discover the QoS that is being experienced by traffic and how to communicate this information in a timely and useful way to an application.

1. **Round Trip Time (RTT)**: The length of time to send a packet to a remote host and back to originator. In general, the lower this value the better – links with a high RTT appear sluggish and unresponsive. However most users will not notice delays around the 10s or possibly 100s of ms range (Abdelzaber and Bhatti, 1999).

2. **Jitter:** The variation in packet delay measurements recorded. Low jitter is better as it indicates a more stable and therefore more predictable link. Jitter is important for interactive, real-time or streaming applications.
3. **Bandwidth**: The amount of data that is transferred per unit time. Together with RTT this impacts on the delay experienced by a client. To use an analogy with water distribution; RTT is the length of the pipe and bandwidth the width.
4. **Packet Loss**: The proportion of packets lost on a link in a given time period. Lower loss values are better, indicating a more efficient use of a link: Lost packets are wasteful as the network resource used to send the lost packet is wasted and further wastage is introduced as a duplicate packet has to be sent.

The approach adopted in this work is to alter the fidelity of media in response to feedback on the level of network congestion, thereby achieving the appropriate trade off between technical quality and delay. Consistency of presentation is achieved by setting the fidelity at the start of a user session. In the absence of sharp and prolonged changes in network QoS, a consistent fidelity will be presented throughout a session.

3.3 Adaptability of Media

When considering the types of media available within Web Pages, they can broadly be categorised into one of several categories, some of which may be adjusted to account for the QoS that a connecting client is receiving. Table 1 provides an overview of the type of resources and the level of QoS adaptation that can be applied to them:

Table 1: QoS Adaptability of Media

Media Type	QoS Adjustable?
Text – HTML, TXT, PDF, etc.	Limited
Graphics – GIF, JPG, TIFF etc.	Highly
Video – AVI, MPEG, DIVX etc.	Highly
Streaming Media	Highly
Active Content – Flash, Java etc	Limited

4 SYSTEM FRAMEWORK

There are three main components to the framework developed: a Network Monitor, a Network Aware Web Server and Content Adaptation Tools. Figure 1 illustrates the Network Monitor positioned so that it is able to monitor all incoming and outgoing traffic, with the resulting QoS data being distributed into a QoS Multicast Zone, where all interested Servers can then receive it; it is possible to have several different Network Aware Servers providing a variety of services to connecting clients, each receiving their QoS information from a single Network Monitoring Agent.

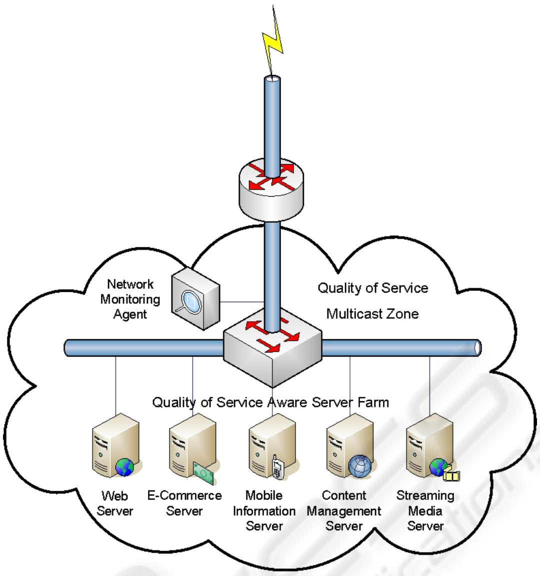
Network Monitoring Agent
Quality of Service Multicast Zone
Quality of Service Aware Server Farm
Web Server
E-Commerce Server
Mobile Information Server
Content Management Server
Streaming Media Server

Figure 1: Network Monitor Positioning

4.1 Network Monitor

The network monitor uses online passive monitoring techniques (Mogul, 1990, Mogul, 1992) to discover the QoS that flows are receiving. Passive monitoring is used in preference to active probes for two reasons. Firstly, no extra wide area traffic is generated which could adversely impact upon the client traffic. Secondly, by extracting measurements from observed traffic more data will be available from the locations where there is the most traffic; predictions of expected network conditions will be more accurate for those locations that use the service most often.

The structure of the network monitor is shown in figure 2 with its design, implementation and evaluation described in more detail in (Ruddle, 2000, Ruddle et al., 2002). TCPDump is used to capture packets, thus allowing the ability to filter traffic and generate trace files for post-mortem analysis. The network monitor extracts congestion information from Transport Control Protocol (TCP) data streams and feeds this information back to end points. Information extraction occurs within a connection layer where per-connection state is maintained. Each reading is then passed to a

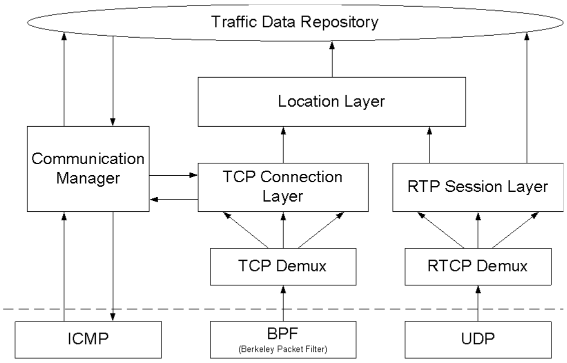

Figure 2: Network Monitor Structure

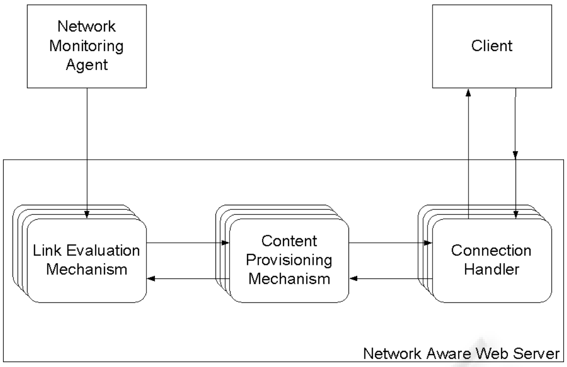

Figure 3: Network Aware Web Server Architecture

location layer where statistics are maintained on a per location basis.

At the connection layer state is maintained for all open connections. As each packet arrives a hash table look up is done. This locates the state for the connection that the packet belongs to. Both TCP Macro-state, which controls the set up and closure of connections, and Micro-state, which controls the flow of data, are monitored. Packet loss is detected by the observation of duplicate acknowledgements, or by the observation of a repeat packet. RTTs are measured by observing the delay between the receipt of an acknowledgement and the observation of the packet that generated the acknowledgement. Only one RTT measurement is attempted per window of data. When a connection finishes all state associated with that connection is reclaimed.

When events occur such as a packet being observed, a loss being discovered or the RTT being measured, these are communicated to the Location Layer. The Location Layer maintains state for aggregates of IP addresses where a Location is defined as an aggregation of hosts to which traffic is likely to experience similar network conditions. State maintained for each location includes the proportion of packets lost, the RTT and the maximum bandwidth observed.

The observation of SYN packets allows new connection requests to be detected and predictions of likely network conditions for that connection to be communicated to the local host in a timely fashion. These predictions are contained within a Location Information Packet. Extensions to the network monitor which support the sharing of congestion information between TCP and Real Time Protocol (RTP) traffic, have been designed (Miller, 2002).

4.2 Network Aware Web Server

It is an important constraint of this architecture that the Network Aware Web Server is able to make QoS decisions automatically whilst working with existing Web Browser software. As such the Network Aware Web Server provides content in the same way as a standard Web Server, fully conforming to the HTTP/1.0 and HTTP/1.1 specifications (Berners-Lee et al., 1996, Fielding et al., 1999).

In the current implementation the Network Aware Web Server supplies different versions of content depending on the QoS characteristics of a link in a transparent and fully automatic manner. Unlike other approaches (Abdelzaber and Bhatti, 1999), adapted resources are not generated on-the-fly as this would take up valuable processing power and introduce extra latency. Instead the Network Aware Web Server chooses between different versions of a resource that are held on backing store. Assuming QoS data relating to a connecting client is available, the Network Aware Web Server implements only marginal changes to the model implemented in a standard Web Server, with the Connection Handler speaking to the Content Provisioning Mechanism and then the Link Evaluation Mechanism before returning the most appropriate resource to the requesting client (figure 3). It is at this stage that indirection is used to send to a client the most suitable resource, based on the quality of the client link. Currently the framework has been developed to support five distinct quality levels – this is done in order to provide distinct sets into which various links can be graded and thus reduce the possible problems caused by a slightly erroneous quality classification; there is less likelihood that small errors will cause a change in grading and so we protect the perceived user quality by reducing unnecessary quality reclassifications, and ensure a more consistent session presentation. As a safeguard, should QoS data not be available the Network Aware Web Server will revert to standard operation, serving the non-QoS adapted version of the requested resource as shown in figure 4.

Although not currently implemented, extensions can be envisioned whereby content providers are able to choose to allow their target audience to

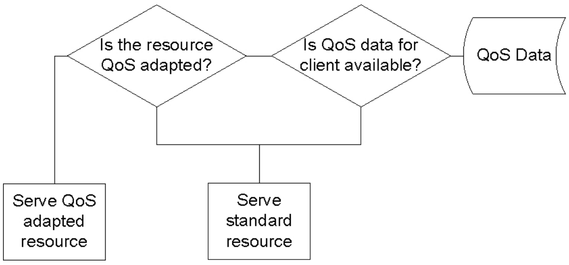
Is the resource QoS adapted?
Is QoS data for client available?
QoS Data
Serve QoS adapted resource
Serve standard resource

Figure 4: QoS Decision Mechanism

control the adaptation mechanisms thereby allowing the user of a resource to determine whether QoS Adaptation is performed or not.

5 RESOURCE MAPPING

Whilst mapping from measured network metrics to a given QoS adapted version of a resource the priority is to minimise the delay experienced by the user whilst maximising the perceived QoS offered. This work takes the view that by more closely matching the amount of information to transfer with the available bandwidth of a client's connection it is possible to maximise user perceived QoS by minimising the delay experienced. In order to achieve this trade-off a mapping function based on the steady state behaviour of TCP is used. As reported by (Mahdavi and Floyd, 1997), the steady state behaviour of TCP provides a benchmark against which the behaviour of other congestion control schemes have been judged. The steady state behaviour of TCP is given by equation 1 (Mathis et al., 1997), where MSS is the maximum segment size on a link, C is a known constant, RTT is the round trip time and P is the probability of loss. Estimates of RTT and P are provided by the LIS, MSS for a connection is known and C is a known constant.

$$T = \frac{MSS * C}{RTT * \sqrt{P}} \qquad (1)$$

If the data flow is application limited low levels of loss may suggest an available bandwidth in excess of the actual physical bandwidth. This can be accounted for by measuring the actual bandwidth utilisation and setting the estimated available bandwidth to the minimum of that estimated from congestion feedback and actual utilisation.

The bandwidth category that a connection request falls into (Q) is then given by equation 2:

$$Q = \min\left(\left[T = \frac{MSS * C}{RTT * \sqrt{P}}\right], \max(B)\right) \qquad (2)$$

In this way the expected available bandwidth for a destination can be calculated based on past measurements. The result may not correspond to the actual access bandwidth, but may rather correspond to the share of bandwidth available on the bottleneck link. Having established the bandwidth available to a destination the Server can then choose the appropriate version of the site to transmit by considering perceptual requirements.

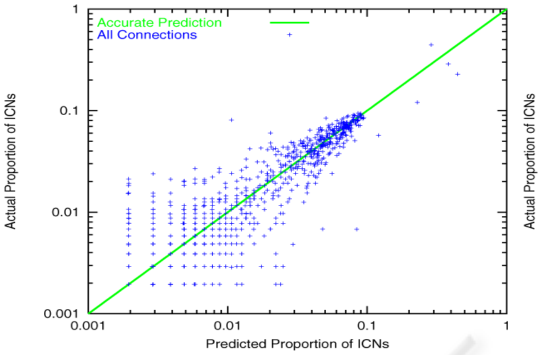

Figure 5: Predicted vs. Actual Congestion

In order to maximise consistency, once an appropriate set of adapted resources have been chosen, the system makes use of the concept of a human session, the aim of which is to provide a consistent fidelity through the lifetime of a session. Here we define a human session as an amount of time spanning a series of requests during which a user might alternate between assimilating the information within a page and downloading new pages. For example a user receives Page 1 from a site and spends 10 minutes reading it, he then proceeds to request Page 2; ideally we want to ensure that the quality of Page 2 is similar to the quality of the Page 1 that has already been received – this will ensure continuity throughout the browsing session. In this situation the Human Session timeout should not be set to anything below 10 minutes. During a session all pages should aim to be served from the same version of the site.

Regardless of any Human Session, changes in the QoS readings are continually tracked during a session. If an initial link quality evaluation is found to be consistently lacking then it may be necessary that remedial action is undertaken, resulting in the link quality being re-graded, in order to provide the most suitable browsing experience to the connecting client. However such a decision should not be taken lightly and cannot be based solely on one, or possibly a handful of bad readings, instead historical data should be considered, with the weight attached to historical data diminishing over time as the data becomes more and more out of date. In order to achieve this data smoothing an Exponentially Weighted Moving Average (EWMA) is used. It also deals with the ageing of historical data, something that the arithmetic mean is unable to do.

6 EVALUATION

The predictive powers of the network monitoring agent has been evaluated using live Internet experiments, between hosts in the US, the UK, Spain, Germany and Eastern Europe. Figure 5 shows the predicted verses the actual levels of congestion experienced. The metric used is the proportion of Implicit Congestion Notifications (ICNs). Packet loss is taken as an implicit indication of congestion. Yet in a single congestion event several packets may be lost, consequently in these experiments only the first packet loss in a window of packets is taken as an ICN.

It is of interest that on links experiencing high levels of congestion the accuracy of prediction increases. Further more it was found that the correlation between prediction and result remained high for periods of time in excess of 20 minutes. These results suggest that it is possible to use the measurement of past traffic to predict the network conditions that are likely to be experienced by future traffic. Consequently the approach adopted here is valid.

7 CONCLUSION

We have presented the design, implementation and evaluation of a framework for making the Web QoS Aware. The approach adopted is to use passive monitoring of transport level headers to make predictions about the QoS that is expected to be experienced by a particular location. A mechanism for translating a single high fidelity Web Site into a set of Web Sites that are appropriate for different bandwidths has been outlined. When a user session starts the Web Server uses QoS information to determine which bandwidth is appropriate. It then serves up Web Pages from the appropriate Web Site. This approach allows Web Designers to leverage the increasing bandwidth many client have available without producing Web Pages that are inaccessible to others.

REFERENCES

Abdelzaber, T. & Bhatti, N., 1999. Web Server QoS Management by Adaptive Content Delivery. International Workshop on Quality of Service. London.

Barnett, C. M., 2002. Adaptive Content Web Serving. New Haven, CT, Yale University.

Berners-Lee, T., Fielding, R. & Frystyk, H., 1996. RFC 1945: Hypertext Transfer Protocol - HTTP/1.0. IETF.

Bhatti, N., Bouch, A. & Kuchinsky, A., 2000. Integrating User-Perceived Quality into Web Server Design. 9th International World Wide Web Conference. Amsterdam.

Bouch, A., Kuchinsky, A. & Bhatti, N., 2000. Quality is in the Eye of the Beholder: Meeting Users' Requirements for Internet Quality of Service. CHI 2000.

Bouch, A. & Sasse, M. A., 1999. Network Quality of Service: What do users need? 4th International Distributed Conference. Madrid.

Bouch, A. & Sasse, M. A., 2000. The Case for Predictable Media Quality in Networked Multimedia Applications. MMCN 2000. San Jose, CA.

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P. & Berners-Lee, T., 1999. RFC 2616: Hypertext Transfer Protocol - HTTP/1.1. IETF.

Floyd, S., Jacobson, V., Liu, C.-G., Mccanne, S. & Zhang, L., 1997. A reliable multicast framework for light-weight sessions and application level framing. IEEE/ACM Transactions on Networking, 5, 784-803.

Jacobson, V., 1988. Congestion Control and Avoidance. ACM Computer Communications Review, 4, 314-329.

Jain, R. & Ramakrishnan, K. K., 1988. Congestion Avoidance in Computer Networks with a Connectionless Network Layer: Concepts, Goals and Methodology. IEEE Computer Networking Symposium. Washington D.C.

Mahdavi, J. & Floyd, S., 1997. TCP-Friendly Unicast Rate-Based Flow Control. Technical note sent to the end2end-interest mailing list: http://www.psc.edu/networking/papers/tcp_friendly.html.

Mathis, M., Semke, J., Mahdavi, J. & Ott, T., 1997. The Macroscopic Behavior of the Congestion Avoidance Algorithm. Computer Communications Review, 27.

Miller, A., 2002. Best Effort Measurement Based Congestion Control. Department of Computer Science. Glasgow, University of Glasgow.

Mogul, J. C., 1990. Efficient use of Workstations for Passive Monitoring of Local Area Networks. ACM SIGCOMM, 20, 253-263.

Mogul, J. C., 1992. Observing TCP Dynamics in Real Networks. ACM SIGCOMM, 22, 305-317.

Pradhan, R. & Claypool, M., 2002. Adaptive Multimedia Content Delivery for Scalable Web Servers. International Network Conference. Plymouth.

Ruddle, A., 2000. A Location Information Server for the Internet. IEEE International Conference on Computer Communications and Networks. Las Vegas, NV.

Ruddle, A., Allison, C. & Lindsay, P., 2002. A Measurement Based Approach to TCP Congestion Control. European Transactions on Telecommunications, 13, 53-64.

Shaha, N., Desai, A. & Parashar, M., 2001. Multimedia Content Adaptation for QoS Management over Heterogeneous Networks. International Conference on Internet Computing. Las Vegas, NV.